



Disciplina: T951 - Sistemas Inteligentes

Professor: André Coelho

Datas de apresentação e entrega do código-fonte: 12/03/2019 e 14/03/2019

Peso da atividade: 4,0

Observação: Vale lembrar que **não** há segunda chamada para avaliações práticas. A **primeira data** será alocada para apresentação dos resultados referentes à rede **Perceptron**, enquanto a **segunda data** para apresentação dos resultados referentes à rede **Adaline**. O **código-fonte** deverá ser entregue ao professor durante as aulas das datas indicadas acima, juntamente com o **relatório técnico** reportando os resultados experimentais. **Todos** os alunos devem estar preparados para responder às perguntas feitas pelo professor, sendo que a **nota** de cada integrante de um grupo será dada com base em seu desempenho individual durante a arguição. **Não** será permitido o uso de *toolkits* ou bibliotecas de redes neurais disponíveis na Internet para implementar os algoritmos. O não-comparecimento implicará em nota **zero** nesta avaliação.

INSTRUÇÕES

Nesta atividade computacional, que poderá ser realizada individualmente ou em grupo de **no máximo 2 integrantes**, deve-se implementar em **Python, Java ou Matlab** os algoritmos de treinamento e teste das redes **Perceptron** e **Adaline**, conforme o material disponibilizado em sala de aula. Os grupos devem se cadastrar enviando mensagem eletrônica com os nomes dos integrantes para acoelho@unifor.br. Para cada grupo, será repassado um *dataset* diferente contendo 100 amostras (padrões) bidimensionais pertencentes a duas classes linearmente separáveis, dentre as quais 80% deverão ser usadas para treinamento e 20% para teste. A partição treino/teste já estará definida para cada *dataset*. A codificação das classes será bipolar. Testes computacionais deverão ser realizados sobre esses dados para cada rede, conforme as tarefas descritas a seguir. Os resultados deverão ser apresentados de forma organizada, alocando uma seção diferente do relatório a cada experimento. **OBS:** o programa deve ter a opção de plotar na tela as amostras de treinamento das classes e as fronteiras de decisão produzidas durante o treinamento das duas redes. Além disso, para o caso da rede Adaline, o programa deve ser capaz de plotar a curva “EQM × época”.

1. PERCEPTRON SIMPLES

Realize os experimentos descritos abaixo separadamente:

- 1) Considerando os dados não-normalizados e a taxa de aprendizado $\eta = 1.0$, realize 5 execuções de treinamento para a rede Perceptron. Na primeira, o vetor de pesos e limiar devem ser iniciados com valores nulos. Nas demais execuções, deve-se iniciar o vetor de pesos e limiar com valores aleatórios entre zero e um. Em cada execução, reinicie o gerador de números aleatórios, de tal forma que as condições iniciais não sejam as mesmas. Anote os valores de pesos e limiar da fronteira de decisão obtida ao final do treinamento.. Também guarde o número de épocas de treinamento. Teste a rede treinada sobre os dados de teste e cheque se todas as amostras de teste são classificadas corretamente.
- 2) Repita o experimento anterior, considerando agora a taxa de aprendizado $\eta = 0.1$.
- 3) Repita o experimento anterior, considerando agora a taxa de aprendizado $\eta = 0.01$.
- 4) Repita os experimentos 1) a 3), considerando agora que os dados estejam normalizados via padronização.
- 5) A partir dos experimentos anteriores, indique qual foi a configuração mais eficiente (ou seja, cujo número médio de épocas de treinamento foi o menor) para treinamento do Perceptron.
- 6) Com base na visualização das amostras de treinamento na tela, selecione uma amostra da classe +1 e outra da classe -1 que estejam bem próximas entre si e altere temporariamente as suas classes. Feita essa alteração, verifique se o processo de treinamento continua convergindo ou não para cada configuração descrita acima.

2. ADALINE

Realize os experimentos descritos abaixo separadamente:

- 1) Considerando os dados normalizados, a taxa de aprendizado $\eta = 1.0$ e a precisão $\varepsilon = 0.1$, realize 5 execuções de treinamento para a rede Adaline no modo off-line (batch). Na primeira, o vetor de pesos e limiar devem ser iniciados com valores nulos. Nas demais execuções, deve-se iniciar o vetor de pesos e limiar com valores aleatórios entre zero e um. Em cada execução, reinicie o gerador de números aleatórios, de tal forma que as condições iniciais não sejam as mesmas. Verifique se em alguma das execuções o algoritmo conseguirá convergir para uma solução adequada. Se isso não ocorrer, explique o motivo.
- 2) Repita o experimento anterior, considerando agora que o treinamento da rede Adaline seja no modo online.
- 3) Repita o experimento 1), considerando agora a taxa de aprendizado $\eta = 0.01$. Anote os valores de pesos e limiar da fronteira de decisão obtida ao final do treinamento. Calcule a taxa de acerto (classificação correta) sobre as amostras de treinamento. Verifique se as fronteiras produzidas pelas diferentes execuções se modificam significativamente ou não. Finalmente, teste a rede treinada sobre as amostras de teste e calcule a taxa de acerto.
- 4) Repita o experimento 3), considerando agora que o treinamento da rede Adaline seja no modo online.
- 5) Repita os experimentos 3) e 4), considerando agora a precisão $\varepsilon = 0.01$.
- 6) Repita os experimentos 3) e 4), considerando agora a precisão $\varepsilon = 0.00001$.
- 7) A partir dos experimentos anteriores, indique qual foi a configuração mais eficiente (ou seja, cujo número médio de épocas de treinamento foi o menor) e qual foi a mais eficaz (ou seja, obteve a maior taxa de acerto sobre as amostras de treinamento e teste).